The Summer Drop in Female Employment: *Review of Economics and Statistics*

Readme for Replication Package

Brendan M. Price
Federal Reserve Board

Melanie Wasserman
UCLA and NBER

April 25, 2024

1 Overview

This replication package reproduces results reported both in the published version of the paper and in the supplementary online appendix. We rely entirely on publicly available data, and we conducted the analysis in MacOS Ventura 13.1 using Stata version 15.1.

To replicate our analysis, users will first need to download the source data from IPUMS, the Bureau of Labor Statistics, and the National Center for Education Statistics. Once the raw data files are in place, running the Stata file `main.do` prepares the data, carries out the analysis, and generates all of the figures and tables that appear in the paper or appendix. Running the code takes about two hours on a 2021 MacBook Pro with an Apple M1 Pro chip and 16 GB of RAM. The raw data files take up 2.7 GB, and data files created by the code take up an additional 2.5 GB.

2 Obtaining the source data

We use free and publicly available data from IPUMS, the Bureau of Labor Statistics, the National Center for Education Statistics, and the US Census Bureau. Since we are not authorized to provide the source data directly, we instead provide detailed instructions for downloading the data and placing it within the project folder. Users will need to create an IPUMS account to download the CPS and ATUS data; the other datasets do not require an account.

Current Population Survey, basic monthly files. Most of our analysis relies on the CPS. From the IPUMS CPS website, we construct our main extract by selecting the CPS basic monthly files for each month from January 1989 through December 2019 and retaining the following variables:

- `YEAR`, `SERIAL`, `MONTH`, `HWTFINL`, `CPSID`, `ASECFLAG`, `MISH`, `STATEFIP`, `MARBASECIDH`, `PERNUM`, `WTFINL`, `CPSIDV`, `CPSIDP`, `AGE`, `SEX`, `RACE`, `MARST`, `POPSTAT`, `MOMLOC`, `MOMLOC2`, `POPLOC`, `POPLOC2`, `SPLOC`, `HISPAN`, `EMPSTAT`, `OCC1990`, `IND1990`, `CLASSWKR`, `UHRSWORKT`, `AHRSWORKT`, `WHYUNEMP`, `WHYABSNT`, `WKSTAT`, `NILFACT`, `EDUC`, `SCHLCOLL`, `MARBASECIDP`, `EARNWT`, `LNKFW1MWT`, `HOURWAGE`, `PAIDHOUR`, `EARNWEEK`, `UHRSWORKORG`, `WKSWORKORG`, `UH_PAYABS_B2`

The resulting extract is named `cps_XXXXX.dta.gz`, where `XXXXX` is the extract number within the user's IPUMS account. Place this file in the project folder under `data/raw/cps/bms`, then update the global macro `$cps_bms` in `code/profile.do` to reflect this extract number.

Current Population Survey, Annual Social and Economic Supplement. From the IPUMS CPS website, we construct an ASEC extract by selecting the 1989–2019 (March) ASEC files and retaining the following variables:

- YEAR, SERIAL, MONTH, CPSID, ASECFLAG, HFLAG, ASECWTH, MARBASECIDH, PERNUM, CPSIDV, CPSIDP, ASECWT, AGE, EMPSTAT, MARBASECIDP, ASECOVERP, OCC90LY, IND90LY, WKSWORK1, UHRSWORKLY, INCWAGE

The resulting extract again takes the form `cps_XXXXX.dta.gz`. Place this file in the project folder under `data/raw/cps/asec`, then update the global macro `$cps_asec` in `code/profile.do`.

American Time Use Survey, person-level extract. From the IPUMS ATUS website, we construct a person-level extract by choosing the rectangular (person) data structure, selecting the 2004–2019 samples, and retaining the following variables:

- Standard variables: RECTYPE, YEAR, CASEID, SERIAL, STRATA, PERNUM, LINENO, MONTH, DAY, WT06, WT20, AGE, SEX, KIDUND18, KIDUND13, KIDUND1, KID1TO2, KID3TO5, KID6TO12, KID13TO17, SCC_OWNHH, DATAQUAL, CPSIDP
- Built-in time-use variables: ACT_CAREHH, ACT_CARENHH, ACT_EDUC, ACT_FOOD, ACT_GOVSERV, ACT_HHACT, ACT_HHSERV, ACT_PCARE, ACT_PHONE, ACT_PROFSERV, ACT_PURCH, ACT_RELIG, ACT_SOCIAL, ACT_SPORTS, ACT_TRAVEL, ACT_VOL, ACT_WORK
- Custom time-use variables constructed by summing time spent on the following activity codes:
  - `basic_childcare`: activities 030101, 030106, 030107, 030108, 030109, 030111, 030199, 030301, 030302, 030303, and 030399.
  - `educ_childcare`: activities 030102, 030201, 030202, 030203, 030204, and 030299.
  - `rec_childcare`: activities 030103, 030104, 030105, and 030110.
  - `travel_childcare`: activities 030112, 180301, 180302, 180303, and 180304.

The resulting extract is `atus_XXXXX.dta`. Place this file in the project folder under `data/raw/atus`, then update the global macro `$atus_person` in `code/profile.do`.

American Time Use Survey, activity-level extract. From the IPUMS ATUS website, we construct an activity-level extract by choosing the hierarchical data structure, selecting the 2004–2019 samples, and retaining the following variables:

- RECTYPE, YEAR, CASEID, SERIAL, STRATA, PERNUM, LINENO, WT06, WT20, ACTLINE, ACTIVITY, DURATION, SCC_OWNHH_LN, START, STOP

The resulting extract is `atus_XXXXX.dat`. In addition, IPUMS provides a Stata command file named `atus_XXXXX.do`. Place both of these files in the project folder under `data/raw/atus`, then update the global macro `$atus_activity` in `code/profile.do`.

Bureau of Labor Statistics. We obtain the non–seasonally adjusted labor force participation rates for men and women aged 25–54 from the Bureau of Labor Statistics. We extract these series from the Labor Force Statistics dataset available at https://download.bls.gov/pub/time.series/ln/ln.data.1.AllData. Place this file in `data/raw/bls`.

National Center for Education Statistics, Schools and Staffing Survey. From the 1999–2000 SASS website, we download data on public school teachers and public school districts. These files are available in ASCII, SAS and SPSS format; we used the program StatTransfer to convert these files to Stata `.dta` format. If StatTransfer is available, place the files `distpub99.dta` and `tchpub99.dta` in `data/raw/sass`. If StatTransfer is not available, users will need to update `code/build/sass_builder.do` to pull the necessary variables directly from the ASCII data.

US Census Bureau state codes and shapefiles. We create a mapping between state names, abbreviations, and FIPS codes using a codebook provided by the US Census Bureau. We create a state-level map of school closure timing using the Census Bureau's 2020 TIGER/Line state shapefile. We provide these files directly in the replication package under the subdirectory `libraries/census`.

3 Running the code

The code is fully automated and requires a minimum of user intervention. It does require internet access, as several files source series from the FRED database via the Stata command `freduse`.

After assembling the source data, open Stata, navigate to the subdirectory `code`, and enter the command `do main.do`. Upon doing so:

1. Stata runs `code/profile.do` with the command-line argument `--install-ados`, which installs custom packages from the `ssc` archive.[1] Of particular note, `profile.do` also declares the global macro `$basepath`, which automatically determines the location of the project folder. The rest of the code uses absolute filepaths beginning with `$basepath`.
2. Stata creates additional subdirectories for data, log files, model estimates, and output files.
3. Stata executes the data preparation scripts in `code/build` in their intended sequence. These files have interdependencies and must be executed in the right order.
4. Stata executes the analysis scripts in `code/share` in alphabetical order. These are standalone files, and they may be executed independently or in any sequence.
5. All figures and tables appear in the subdirectory `output`.

`main.do` also accepts four optional command-line arguments, which can be invoked by entering the command `do main.do [argument(s)]`. These are:

- `--no-install`: skips downloading of custom packages from `ssc`.
- `--no-build`: skips execution of the scripts in `code/build`.
- `--no-share`: skips execution of the scripts in `code/share`.
- `--no-estimate`: sets the global macro `$estimate` equal to 0. To speed execution, most of the files in `code/share` save their results within the subdirectory `models`. Once the results have been saved to disk, setting `$estimate equal to 0` tells Stata to skip the (slower) data preparation and estimation steps and proceed immediately to plotting or reporting the results. As one use case, this structure makes it easy to adjust figure aesthetics without rerunning an analysis.

[1] Stata always runs a file called `profile.do` upon launch. Opening Stata within the `code` folder causes Stata to run the local version of `profile.do`, but without installing these packages.

Almost everything is done within Stata, and the code is compatible with Stata versions 15.1 or higher. The code relies on MacOS utilities in three places:

- `code/build/atus_builder.do` uses the shell utility `gunzip` to unzip an ATUS extract provided by IPUMS in `.gz` format. If `gunzip` is unavailable, users can unzip this file manually.
- `code/share/calendars.do` uses the shell command `sips` to convert a large `.pdf` file to a smaller `.png` format. If needed, this step can be skipped.
- `libraries/nicepdf.ado` exports graphs to `.pdf` format; its purpose is to facilitate the use of LaTeX fonts and standardized figure dimensions. `nicepdf` relies on the shell utilities `epstopdf` and `pdfcrop`. Omitting the option `indirect` removes the dependence on `epstopdf` but may prevent the custom fonts from rendering correctly. Adding the option `nocrop` removes the dependence on `pdfcrop`.

`profile.do` installs the following packages available through `ssc`: `addplot`, `coefplot`, `estout`, `filelist`, `ftools`, `freduse`, `gtools`, `gzsave`, `ipfraking`, `ivreg2`, `labutil`, `ranktest`, `regsave`, `shp2dta`, `spmap`, and `xlincom`. We also use the package `grc1leg`. In addition, `profile.do` sets the `adopath` to draw exclusively on `.ado` files located within the project directory. Doing so avoids any inadvertent external dependencies on custom programs not residing inside this project, and it ensures that replicators will run the code using the same version of each package that we used in our analysis.

4 List of files (in order of execution)

Main programs (within `code`):

- `code/main.do`: Executes all Stata programs in sequence.
- `code/profile.do`: Configures project settings and installs packages.

Data preparation (`code/build`):

- `geo_builder`: Prepares geographic crosswalks and files.
- `turning_points.do`: Identifies knots for the linear spline we use as a regression control. (The knots are hardcoded into other files; `turning_points.do` shows how we chose them but doesn't produce output used directly by other files.)
- `cps_bms_builder.do`: Assembles and processes CPS basic monthly files.
- `cps_bms_sampler.do`: Creates our main CPS estimation sample, which it saves in zipped format as `cps_bms_sample.dta.gz`.
- `cps_bms_raking.do`: Augments our CPS sample with raked longitudinal weights that ensure consistency between labor market stocks and flows.
- `cps_asec_builder.do`: Processes the CPS ASEC data.
- `atus_builder.do`: Processes the ATUS data.
- `sass_builder.do`: Processes the SASS data.
- `create_occ1990_2d.do`: Aggregates 3-digit IPUMS occupation codes to the 2-digit level. This file is executed by two other `.do` files but is not executed directly by `main.do`.

Data analysis (`code/share`):

- `annuals.do`: Compute women's counterfactual EPOP and LFPR were these rates to hold at their May levels during the summer months.
- `between.do`: Show the contribution of sectoral and occupational sorting to the gender gap in summer employment.
- `calendars.do`: Show cross-state differences in the timing of school closures.
- `careers.do`: Show how mothers' propensity to work in the education sector varies with the age of their youngest child.
- `couples.do`: Show seasonality in joint employment patterns within married couples.
- `edemp.do`: Show seasonality in education-sector employment in the CPS and CES.
- `edexits.do`: Show gender differences in education workers' propensity to exit employment over the summer.
- `evolution.do`: Show how the summer drop in female employment has changed over our analysis period.
- `explain_demogs.do`: Identify contributors to demographic heterogeneity in the summer drop.
- `heterogeneity.do`: Show heterogeneity in the summer drop by age, race/ethnicity, and educational attainment.
- `hhstatus.do`: Show heterogeneity in the summer by marital status and parental status.
- `job_decomp.do`: Decompose gender differences in summer employment into contributions from sorting across job types and gender differences in employment conditional on job type.
- `life_cycle.do`: Show how the summer drop varies over the life cycle.
- `logs_levels.do`: Show seasonality in women's hours and earnings in both logs and levels.
- `margins.do`: Decompose the summer drop in women's hours into contributions along the extensive and intensive margins.
- `motivation.do`: Show the raw time series for non–seasonally adjusted male and female LFPR.
- `multiple_kids.do`: Show how the summer drop among parents varies with their number of children.
- `occ_flows.do`: Show which occupations contribute to the summer drop.
- `overall.do`: Show seasonality in employment, unemployment, and non-participation.
- `reasons.do`: Decompose summer increases in unemployment and non-participation by reason for unemployment and major activity while out of the labor force.
- `recurrence.do`: Show seasonality in repeat separations from employment occurring 12 months apart.
- `sass.do`: Analyze gender differences in teachers' summer earnings.

- `sorting.do`: Show women's sorting into the education sector within occupations represented in both education and non-education, and calculate earnings penalties associated with education-sector jobs.
- `stocks_flows.do`: Decompose the summer drop into contributions from elevated outflows and depressed inflows.
- `summer_layoffs.do`: Show which sectors drive the summer increase in temporary layoffs.
- `sumstats.do`: Compute summary statistics for our main CPS estimation sample.
- `synchrony.do`: Show that the timing of the summer drop coincides with cross-state differences in the timing of school closures.
- `timeuse.do`: Show seasonality in time spent on childcare.
- `twomonth.do`: Show seasonality in the prevalence of non-work spells encompassing multiple CPS reference weeks.
- `unconnected.do`: Show the summer drop among women with no observed affiliation with the education sector.
- `unpaid_leave.do`: Show seasonality in paid versus unpaid absences.
- `vacation.do`: Show seasonality in vacation-related absences.
- `winter_sectors.do`: Show which sectors drive the winter drop in employment.

Custom Stata programs (`libraries/ado/custom`):

- `classify_jobs.ado`: Classify occupation–industry pairs into a set of coarse job types.
- `load_estimates.ado`: Load saved estimates into memory.
- `make_coeflabels.ado`: Create month labels for regression estimates.
- `nicepdf.ado`: Export figures with desired scaling, fonts, and cropping.
- `process_estimates.ado`: Process and save regression estimates.

Other supporting files (`libraries`):

- `census/tl_2020_us_state/`: State-level shapefile from the Census Bureau.
- `census/state.txt`: List of state FIPS codes from the Census Bureau.
- `fonts/`: Computer Modern fonts, which we redistribute under the terms of the SIL Open Font License, Version 1.1. A copy of that license is included alongside the font files.
- `schemes/`: Custom scheme `spartan` developed by coauthor Brendan Price for producing publication-quality Stata graphics.

5 Codebook for main CPS sample

Our main CPS sample, which is saved as `data/derived/cps_bms_sample.dta.gz`, is created by `code/build/cps_bms_sampler.do` and contains the following variables.

Variable name	Definition
pid	CPS ID, person record
hid	CPS ID, household record
tm	Year/month
year	Calendar year
month	Calendar month
pernum	Person number in sample unit
mish	Month in sample, household level
wtfinl	Final basic weight
wtraked	Raked longitudinal weights (reconciling employment stocks/flows)
marbasecidp	Unique ID for linking March basic to ASEC
tmspline1	Linear spline in calendar time
tmspline2	Linear spline in calendar time
tmspline3	Linear spline in calendar time
tmspline4	Linear spline in calendar time
tmspline5	Linear spline in calendar time
tmspline6	Linear spline in calendar time
tmspline7	Linear spline in calendar time
weeks	Number of weeks elapsed since previous month's reference week
linked_monthly	Valid longitudinal link relative to previous month
linked_complete	Valid longitudinal links across all observations
female	Female
age	Age
age_bin	Age (5-year bins)
wbho	Race and ethnicity
educ	Educational attainment
state_fips	State FIPS code
school_status	Attendance at high school or college/university
marstat	Marital status
spouse_present	Married with the spouse present
hhstatus	Household structure
sploc	Person number of spouse
ownkids	Number of own children under 18 present in household
ownkids_age0	Number of own children age 0 in household
ownkids_age1	Number of own children age 1 in household
ownkids_age2	Number of own children age 2 in household
ownkids_age3	Number of own children age 3 in household
ownkids_age4	Number of own children age 4 in household
ownkids_age5	Number of own children age 5 in household
ownkids_age6	Number of own children age 6 in household
ownkids_age7	Number of own children age 7 in household
ownkids_age8	Number of own children age 8 in household
ownkids_age9	Number of own children age 9 in household

ownkids_age10	Number of own children age 10 in household
ownkids_age11	Number of own children age 11 in household
ownkids_age12	Number of own children age 12 in household
ownkids_age13	Number of own children age 13 in household
ownkids_age14	Number of own children age 14 in household
ownkids_age15	Number of own children age 15 in household
ownkids_age16	Number of own children age 16 in household
ownkids_age17	Number of own children age 17 in household
youngest	Binned age of youngest own child in the household
youngest_years	Age of youngest own child under 18 in the household
emp	Employed
unemp	Unemployed
nlf	Not in labor force
lfp	Labor force participant (== 1 - nlf)
ftemp	Full-time employed last week
ptemp	Part-time employed last week
absent	Employed, not at work last week
empstat	Employment status
hours	Actual hours worked last week (= 0 if not at work)
occ1990	3-digit occupation, 1990 basis
occ1990_2d	2-digit occupation, 1990 basis
ind1990	3-digit industry, 1990 basis
school	Industry is educational services
teacher	Occupation is teaching
whyunemp	Reason for being unemployed
whynilf	Reason for being (or major activity while) out of the labor force
whyabsnt	Reason for not being at work
paid_absence	Paid for time off last week (if absent)
earnweek	Weekly earnings, Dec. 2019 $
earnwt	Earnings weight
earn_sample	Included in earnings analyses
earnings	Estimated earnings
selfemp	Indicator for being self-employed (populated only for the employed)
hourly	Paid by the hour
hourwage	Hourly wage, Dec. 2019 $
uhrsworkt	Hours usually worked per week

6 List of figures and tables

The code reproduces all figures and tables appearing in either the published manuscript or the supplementary online appendix. The table below provides the mapping between output file names and the figure and table numbers as they appear in the paper.

Figure/table #	Program (in `code/share/`)	Output file (in `output/`)
Figure 1	`motivation.do`	`motivation.pdf`
Figure 2	`overall.do`	`overall.pdf`
Figure 3	`stocks_flows.do`	`stocks_flows.pdf`
Figure 4	`logs_levels.do`	`logs_levels_hours.pdf`

Figure 5	synchrony.do	synchrony_epop_women.pdf
Figure 6	hhstatus.do	hhstatus_women.pdf
Figure 7	reasons.do	reasons_women.pdf
Figure 8	timeuse.do	timeuse_6to12.pdf
Figure 9a	edexits.do	edexits_nonemp.pdf
Figure 9b	synchrony.do	synchrony_noned_sephaz.pdf
Figure 10	job_decomp.do	job_decomp_emp.pdf
Figure 11	logs_levels.do	logs_levels_earn1.pdf
Figure 12a	sorting.do	sorting_char_femsh.pdf
Figure 12b	sorting.do	sorting_penalties_earnings.pdf
App. Fig. A1	evolution.do	evolution.pdf
App. Fig. A2	heterogeneity.do	heterogeneity.pdf
App. Fig. A3	winter_sectors.do	winter_sectors.pdf
App. Fig. A4	recurrence.do	recurrence.pdf
App. Fig. A5	unpaid_leave.do	unpaid_leave.pdf
App. Fig. A6	twomonth.do	twomonth.pdf
App. Fig. A7	calendars.do	calendars_map.pdf
App. Fig. A8	calendars.do	calendars_scatter.pdf
App. Fig. A9	synchrony.do	synchrony_epop_men.pdf
App. Fig. A10	life_cycle.do	life_cycle.pdf
App. Fig. A11	multiple_kids.do	multiple_kids_women.pdf
App. Fig. A12	hhstatus.do	hhstatus_men.pdf
App. Fig. A13	multiple_kids.do	multiple_kids_men.pdf
App. Fig. A14	couples.do	couples.pdf
App. Fig. A15	reasons.do	reasons_men.pdf
App. Fig. A16	summer_layoffs.do	summer_layoffs.pdf
App. Fig. A17	timeuse.do	timeuse_und6.pdf
App. Fig. A18	vacation.do	vacation.pdf
App. Fig. A19	edemp.do	edemp.pdf
App. Fig. A20	careers.do	careers.pdf
App. Fig. A21	edexits.do	edexits_nohours.pdf
App. Fig. A22	unconnected.do	unconnected.pdf
App. Fig. A23	job_decomp.do	job_decomp_atw.pdf
App. Fig. A24	sass.do	sass.pdf
App. Fig. A25	sorting.do	sorting_char_ptemp.pdf
App. Fig. A26	sorting.do	sorting_penalties_wage.pdf
App. Tab. A1	sumstats.do	sumstats.tex
App. Tab. A2	margins.do	margins.tex
App. Tab. A3	explain_demogs.do	explain_demogs.tex
App. Tab. A4	between.do	between.tex
App. Tab. A5	job_decomp.do	job_decomp_emp.tex